CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

We live in a constantly changing world, and consequently our beliefs have to be revised whenever there is new information. When we encounter a new piece of information that contradicts our current beliefs, we revise our beliefs rationally.

In the last three decades, the field of computer science has grown substantially beyond mere number crunching, and aspires to imitate rational thinking of human beings. A separate branch of study, artificial intelligence (AI) has evolved, with a number of researchers attempting to represent and manipulate knowledge in a computer system. Much work has been devoted to study the statics of the knowledge, i.e. representing and deducting from fixed knowledge, resulting in the development of expert systems. The field of logic programming, conceived in last seventies, has proved to be an important tool for handling static knowledge. However, such fixed Horn knowledge based systems can not imitate human thinking, unless they are accomplish revising their knowledge in the light of new information. As mentioned before, this revision has to take place rationally. This has led to a completely new line of research, the dynamics of belief.

Studies in dynamics of belief are twofold: What does it mean to rationally revise a belief state? How can a belief state be represented in a computer and revised? The first question is more philosophical theory, and a lot of works have been carried out from epistemological perspective to formalize belief dynamics. The second question is computation oriented, and has been addressed differently from various perspectives of application. For example, a lot of algorithms have been proposed in logic programming to revise a Horn knowledge base or a database represented as a logic program; number of algorithms are there to carry out a view update in a rational database; algorithm to carry out diagnosis; algorithm for abduction reasoning and so on. We need the concept
of "change" in some form or other and thus need some axiomatic characterization to ensure that the algorithms are rational. Unfortunately, till this date, these two tracks remain separate, with minimal sharing of concepts and results. The primary purpose of the dissertation is to study these two developments and integrate them.

When a new piece of information is added to a Horn knowledge base (Delgrande 2008 and Delgrande & Peppas 2011), (Papini 2000) it may become inconsistent. Revision means modifying the Horn knowledge base in order to maintain consistency, by keeping the new information and removing the least possible previous information. In our case, update means revision and contraction, that is insertion and deletion in database perspective. Previous works (Aravindan & Dung 1994), (Aravindan 1995) have explained connections between contraction and knowledge base dynamics. Our Horn knowledge base dynamics is defined in two parts: an immutable part (Horn formulae) and updatable part (literals) (for definition and properties see the works of Nebel 1998, Segerberg 1998, Hansson et al 2001 and Fermé & Hansson 2001). Knowledge bases have a set of integrity constraints. In the case of finite knowledge bases, it is sometimes hard to see how the update relations should be modified to accomplish certain Horn knowledge base updates.

1.2 MOTIVATION

In the general case of arbitrary formulae, the revision problem for knowledge bases is hard to solve. So we restrict the revision problem to Horn formulae. The connection between belief change and database change is an interesting one since so far the two communities have independently considered two problems that are very similar, and our aim is to bring out this connection.

We aim to bridge the gap between philosophical and database theory. In such a case, Hansson’s (Hansson 1997) kernel change is related to the abductive method. Aliseda’s (Aliseda 2006) book on abductive reasoning is one of our key motivation. Wrobel’s (Wrobel 1995) definition of first-order theory revision was helpful to frame our algorithm. On the other hand, we are dealing with the view update problem. Keller and
Minker’s (Keller 1985 and Minker 1996) work is one the motivation for the view update problem. In Figure 1.1 understand the concept of view update problem in rational way. Figure show that foundation form Belief Revision theory, intermediate step handle to Horn knowledge base, this step very impairment that agent have background knowledge and he/she made decision with postulate may require to process next step. Target of the application is connect database updates via Horn knowledge base with abduction reasoning. All clear procedure shown in each section.

![Figure 1.1: Layout of the thesis work](image)

Following example illustrates the motivation of the thesis:

**Example 1.2.1.** Consider a database with an (immutable) rule that a staff member is a person who is currently working in a research group under a chair. Additional (updatable) facts are that matthias and gerhard are group chairs, and delhibabu and aravindan are staff members in group infor1. Our first integrity constraint (IC) is that each research group has only one chair ie. \( \forall x, y, z \ (y=z) \leftarrow \text{group_chair}(x,y) \land \text{group_chair}(x,z) \). Second integrity constraint is that a person can be a chair for only one research group ie. \( \forall x, y, z \ (y=z) \leftarrow \text{group_chair}(y,x) \land \text{group_chair}(z,x) \).

**Immutable part:** \( \text{staff_chair}(x,y) \leftarrow \text{staff_group}(x,z),\text{group_chair}(z,y) \).

**Updatable part:** \( \text{group_chair}(\text{infor1},\text{matthias}) \leftarrow \)
Suppose we want to update this database with the information, staff_chair(aravindan,gerhard); From the immutable part, we can deduce that this can be achieved by asserting
\[
\text{staff}_\text{group}(aravindan,z) \land \text{group}_\text{chair}(z,\text{gerhard})
\]

If we are restricted to definite clauses, there are three plausible ways to do this: first case is, aravindan and gerhard belong to infor1, i.e, \text{staff}_\text{group}(aravindan,infor1) \land \text{group}_\text{chair}(infor1,gerhard). We need to delete both base facts \text{group}_\text{chair}(infor1,matthias)← and \text{group}_\text{chair}(infor2,gerhard)←, because our first IC as well as second IC would be violated otherwise. In order to change the view, we need to insert \text{group}_\text{chair}(infor1,gerhard)← as a base fact. Assume that we have an algorithm that deletes the base facts \text{staff}_\text{group}(delhibabu,infor1)← from the database. But, no rational person will agree with such an algorithm, because the fact \text{staff}_\text{group}(delhibabu,infor1)← is not "relevant" to the view atom.

Second case, aravindan and gerhard belong to infor2, that is \text{staff}_\text{group}(aravindan,infor2) \land \text{group}_\text{chair}(infor2,gerhard). Simply, insert the new fact \text{staff}_\text{group}(aravindan,infor2)← to change the view. Suppose an algorithm deletes the base facts \text{staff}_\text{group}(aravindan,infor1)← from the database, then it can not be "rational" since these facts are not "relevant" to the view atom.

Third case, aravindan and gerhard belong to infor3 (free assignment of the group value), that is \text{staff}_\text{group}(aravindan,infor3) \land \text{group}_\text{chair}(infor3,gerhard). Suppose, we insert new base fact \text{group}_\text{chair}(infor3,gerhard) ←, our second IC does not follow. Suppose an algorithm inserts the new base fact \text{staff}_\text{group}(aravindan,infor2)← or \text{staff}_\text{group}(aravindan,infor1)← is deleted, then it can not be "rational".

The above example highlights the need for some kind of "relevance policy" to be adopted when a view atom is to be inserted to a deductive database. How many such axioms and policies do we need to characterize a "good" view update? When are we sure that our algorithm for view update is "rational"? Clearly, there is a need for an
axiomatic characterization of view updates. By axiomatic characterization, we mean explicitly listing all the rationality axioms that are to be satisfied by any algorithm for view update.

The basic idea in (Behrend & Manthey 2008), (Aravindan & Baumgartner 1997) is to employ the model generation property of hyper tableaux and magic set to generate models, and read off diagnosis from them. One specific feature of this diagnosis algorithm is the use of semantics (by transforming the system description and the observation using an initial model of the correctly working system) in guiding the search for a diagnosis. This semantical guidance by program transformation turns out to be useful for database updates as well. More specifically we use a (least) Herbrand model of the given database to transform it along with the update request into a logic program in such a way that the models of this transformed program stand for possible updates.

We discuss two ways of transforming the given database together with the view update (insert and delete) request into a logic program resulting in two variants of view update algorithms. In the first variant, a simple and straightforward transformation is employed. Unfortunately, not all models of the transformed program represent a rational update using this approach. The second variant of the algorithm uses the least Herbrand model of the given database for the transformation. In fact what we referred to as offline preprocessing before is exactly this computation of the least Herbrand model. This variant is very meaningful in applications where views are materialized for efficient query answering. The advantage of using the least Herbrand model for the transformation is that all models of the transformed logic program (not just the minimal ones) stand for a rational update.

When dealing with the revision of a Horn knowledge base (both insertions and deletions), there are other ways to change a Horn knowledge base and it has to be performed automatically also (Fermé 1992 and Rodrigues & Benevidas 1994). Considering the information, change is precious and must be preserved as much as possible. The principle of minimal change (Gärdenfors 1998, Dalal 1988 and Herzig & Rifi 1999), (Schulte 1999) can provide a reasonable strategy. On the other hand, practical implementations have to handle contradictory, uncertain, or imprecise information, so several prob-
lems can arise: how to define efficient change in the style of Carlos Alchourrón, Peter Gärdenfors, and David Makinson (AGM) (Alchourron et al. 1985b); what result has to be chosen (Lakemeyer 1995), (Lobo & Trajcevski 1997), (Nayak et al. 2006); and finally, according to a practical point of view, what computational model to explore for the Horn knowledge base revision has to be provided?

1.3 RELATED WORKS

We begin by recalling previous work on view deletion. Aravindan (Aravindan & Dung 1994), (Aravindan 1995), defines a contraction operator in view deletion with respect to a set of formulae or sentences using Hansson’s (Hansson 1997a) belief change. Similar to our approach, he focused on set of formulae or sentences in knowledge base revision for view update with respect to insertion and deletion and formulae are considered at the same level. Aravindan proposed different ways to change knowledge base via only database deletion, devising particular postulate which is shown to be necessary and sufficient for such an update process.

Our Horn knowledge base consists of two parts, immutable part and updatable part, but our focus is on minimal change computations. The related works are, Eiter (Eiter & Makino 2007), Langlois (Langlois et al. 2008) and Delgrande (Delgrande & Peppas 2011) are focus on Horn revision with different perspectives like prime implication, logical closure and belief level. Segerberg (Segerberg 1998) defined a new modeling technique for belief revision in terms of irreversibility on prioritized revision. Hansson constructed five types of non-prioritized belief revision. Makinson (Makinson 1997) developed dialogue form of revision AGM. Papini (Papini 2000) defined a new version of knowledge base revision. In this thesis, we considered the immutable part as a Horn clause (Fermé & Hansson 2001 shown shielded contraction similar to immutable part, the success postulate does not hold in general; some non-tautological beliefs are shielded from contraction and cannot be given up. Shielded contraction has close connections with credibility limited revision shown Hansson et al 2001) and the updatable part as an atom (literal). Knowledge bases have a set of integrity constraints.
Hansson’s (Hansson 1997a) kernel change is related to abductive method. Aliseda’s (Aliseda 2006) book on abductive reasoning is one of the motivation step. Christiansen’s (Christiansen & Dahl 2009) work on dynamics of abductive logic grammars exactly fits our minimal change (insertion and deletion). Wrobel’s (Wrobel 1995) definition of first order theory revision was helpful to frame our algorithm.

On the other hand, we are dealing with view update problem. Keller’s (Keller 1985) thesis is motivation of the view update problem. There are many papers related to the view update problem (for example, the recent survey paper on view update by Chen and Liao (Chen & Liao 2010) and the survey paper on view update algorithms by Mayol and Teniente (Mayol & Teniente 1999). More similar to our work is the paper presented by Bessant (Bessant et al. 1998), which introduces a local search-based heuristic technique that empirically proves to be often viable, even in the context of very large propositional applications. Laurent (Laurent et al. 1998), considers updates in a deductive database in which every insertion or deletion of a fact can be performed in a deterministic way.

Furthermore, and at a first sight more related to our work, some work has been done on "core-retainment" (Hansson 1991) in the model of language splitting introduced by Parikh (Parikh 1999). More recently, Doukari (Doukari et al. 2008), Özçep (Özçep 2012) and Wu (Wu et al. 2011) applied similar ideas for dealing with knowledge base dynamics. These works represent motivation step for our future work. Second, we are dealing with how to change minimally in the theory of "principle of minimal change", but current focus is on finding second best abductive explanation (Liberatore & Schaerf 2004 and 2012), 2-valued minimal hypothesis for each normal program (Pinto & Pereira 2011). Our work reflected in the current trends on Ontology systems and description logics (Qi and Yang (Qi & Yang 2008) and Kogalovsky (Kogalovsky 2012)). Finally, when we presented Horn knowledge base change in abduction framework, we did not talk about compilability and complexity (see the works of Liberatore (Liberatore 1997) and Zanuttini (Zanuttini 2003)).
1.3.1 Objective

The main objectives of this thesis are:

- To define a new kind of revision operator on Horn knowledge base and obtain axiomatic characterization for it.
- To propose new generalized revision algorithm for Horn knowledge base dynamics, and study its connections with kernel change and abduction procedure.
- To develop a new view insertion algorithm for databases.
- To design a new view update algorithm for stratifiable Deductive Database (DDB), using an axiomatic method based on Hyper tableaux and magic sets.
- To study an abductive framework for Horn knowledge base dynamics.
- To present a comparative study of view update algorithm and integrity constraint.

1.4 ORGANIZATION OF THIS DISSERTATION

The rest of the dissertation is organized as follows: Chapter 2 is a review of concepts form belief dynamics proposed by AGM, and others. In chapter 3, we introduce knowledge base dynamics along with the concept of generalized revision, and revision operator for knowledge base. Chapter 4 studies the relationship between knowledge base dynamics and abduction. We discuss an important application of knowledge base dynamics in providing an axiomatic characterization for update view literals over databases. We briefly discuss hyper tableaux calculus and magic set in Chapter 5. In chapter 6, we provide an abductive framework for Horn knowledge base dynamics in first order version. In chapter 7, we draw conclusions with a summary of our contribution and indicate future directions of our investigation. A comparative study of view update algorithm and integrity constraint with our axiomatic method can be found in the Appendix.